The Dorm Room Genie

The Door Lock

--To be completed by David Nedzel

Students are issued keys by which to access their rooms, but far too often a key is forgotten inside the room leaving someone locked out. After a lockout, a student must go to the front desk and obtain the spare key. If the spare key was locked inside as well, the student must call someone to come over and open their door. Needless to say, if you locked yourself out during a late-night bathroom run, this is a little inconvenient.

The door lock allows the student to set an alphanumeric access code which will release the door latch and let the student enter. The student will be able to set their own access code, and change it whenever they choose. The lock can also be set to "unlock" mode. This will allow a student to leave the door unlocked if they will be frequently leaving or entering, or if they wish to allow anyone to come in while they are inside. The lock will also have an "inactive" mode. This mode will allow the user to disable the keypad entry while they are inside. This will provide an extra layer of security as the only way to gain entry at that point would be to obtain the spare key. This feature could only be turned on while the student is inside via the inside user interface and would be automatically disabled if the student opens the door. This will prevent the student from accidentally disabling the keypad and locking themselves out.

Additional Parts Required: Electronic Latch (but recognizing that this is an expensive item, often not useful to 6.111 students, we'll use an LED to indicate the status "locked" or "unlocked" of the latch.)

Inputs: time, 8-bit wide alpha-numeric keystroke input, door-sensor, lock, unlock, inactive, change password

Outputs: lock-state, lock control

The door lock has several internal states:
1. Unlocked: In this state, the user may transition to locked, inactive, or password change mode by pressing the appropriate key.
2. Locked: In this state, the user may transition to unlocked, inactive or password change mode by pressing the appropriate key. The user may also enter the entry state by typing in the current password
3. Entry: In this state the lock will unlock for 5 seconds, or until the door opens, whichever is shorter, and then return to the locked state. The user may also transition to locked, unlocked, inactive or password change by pressing the appropriate key.
4. Inactive: In this state, the user may transition to locked, inactive or password change mode by pressing the appropriate key. By opening the door, the state will automatically transition to "locked".
5. Password change: In this state, the lock will accept a new password after prompting the user for the old password. After the password change sequence is complete, the lock returns to the "locked" state.

The Burglar Alarm System
--To be completed by David Nedzel

The Dorm Room Genie also implements a burglar alarm. The alarm provides additional security by sounding an alarm when the door is opened. This can be useful both when the student is home, and away from their room. While at home, a sleeping student will be awoken, and able to intercept the intruder. If away, friends nearby could respond.

Additional Parts Required: magnetic sensor switch, alarm buzzer

Inputs: time, door-sensor, lock-mode, lock-state, wake-up, arm, disarm

Outputs: burglary, burglar alarm-state

The burglar alarm has several states:

1. Disarmed: The alarm can transitions to armed or exit when the user pushed arm depending upon whether the door is open or closed.
2. Armed: The alarm can transition to burglary if the door is opened, or disarmed if the if the appropriate key is pressed.
3. Exit: The alarm transitions to armed when the door closes
4. Burglary: the alarm transitions to disarmed when the password is entered, or else armed once 2 minutes have elapsed and the door is closed.

Extension: If we have additional time, we might implement a feature to inform the student of any intrusions and when they occurred once the alarm has been disabled.

Clock
--To be completed by Aaron Stonely

The clock is designed to keep the system time. The clock will have the basic features of a standard digital watch. The clock will take the 27mghz internal clock and convert it to seconds. The clock module will then use the seconds and the time set by the user to keep the current time. The clock will be used by the burglar alarm, lighting, and alarm clock modules, so that they may use it to implement certain functionality. The clock will output the time to the display module, so that it can be displayed to the user.

Inputs: set time

Outputs: time

Wakeup Alarm
--To be completed by Aaron Stonely

The alarm clock module will implement additional functionality to the clock. The alarm clock allows the student to be gently awoken by light, even if their curtains are closed. The lights will gradually brighten, reaching their brightest level at the pre-programmed time. If the system is not deactivated, the lights will begin to flash, and eventually, the burglar alarm will sound in shorts bursts if the student fails to acknowledge the wake-up sequence.

Additional parts required: Buzzer (same buzzer as burglar alarm)

Inputs: set wake-up alarm, alarm acknowledge, snooze, time

Outputs: alarm, buzzer, wake-up time, wake-up state;

The alarm clock has several states:

1. Off: If the alarm clock is not set, it will not go off.
2. Set: In this state, the user has set the time to wake up and has requested a wakeup, but the time has not yet been reached. The alarm clock continues to wait until the current time reaches the lighting threshold time before transitioning states.
3. Wake-up sequence: At this point, the current time is within the wake-up time threshold and the lights will slowly begin to turn on, reaching full intensity at the set time.
4. Active: After the programmed wake up time has passed, the system waits a few minutes before beginning to beep the buzzer.
5. Snooze: The lights will dim briefly and the buzzer will be silenced in order to give the user a few extra minutes of sleep.

The Lighting Control

--To be completed by Aaron Stonely

The lighting dimming control circuitry can be set to “off”, “manual”, “automatic”, and “sleep” mode. When in “off” mode, the lights will remain off. In “manual” mode, the user can manually adjust the brightness of the lights. In “automatic” mode, the lights will automatically adjust their brightness based upon input from a photo-electric sensor. As the sun sets, or lighting is otherwise reduced, they will turn themselves on. The lighting module also interfaces with other modules. When the student returns home, the lights will switch from “off” to “automatic” mode. The lighting module also interfaces with the burglar alarm, flashing the lights when an unauthorized entry is detected. Finally, the lighting module can be put in “sleep” mode. During “sleep” mode, the lights remain off, even if the student returns home during while it is dark, —an important feature for middle-of-the-night bathroom visits. While in “sleep” mode, the light control circuitry also interacts with the alarm clock as described below.

Additional Parts Required: photo-electric sensor, digitally controlled dimming module for a lamp.

Inputs: clock, manual lights, automatic lights, sleep-mode, lights on, lights off, dim, brighten, burglary, wake-up, photoelectric sensor

Outputs: lighting state, light control

The lighting control has several states:

1. Off: In this state, the lights remain off and automatic adjustment by system is disabled.
2. Manual: The user can adjust the brightness manually through a keyboard command. When in this state, automatic operation is disabled.
3. Automatic: When in this state, light intensity is controlled by the system. As it gets darker, the light intensity is increased.
4. Sleep mode: In this state, the system enters sleep mode on request by the user. The lights remain off and automatic operation is suspended until the alarm clock has gone off, then automatic mode is resumed.

The Keyboard
--To be completed by David Nedzel

In order for the user to interact with the system, there must be a way for the user to provide input to the system. This is accomplished with a keyboard. The keyboard module interoperates keystrokes, synchronizes and debounces them, and passes the appropriate signals onto all the modules of the system.

Input: PS2 keyboard

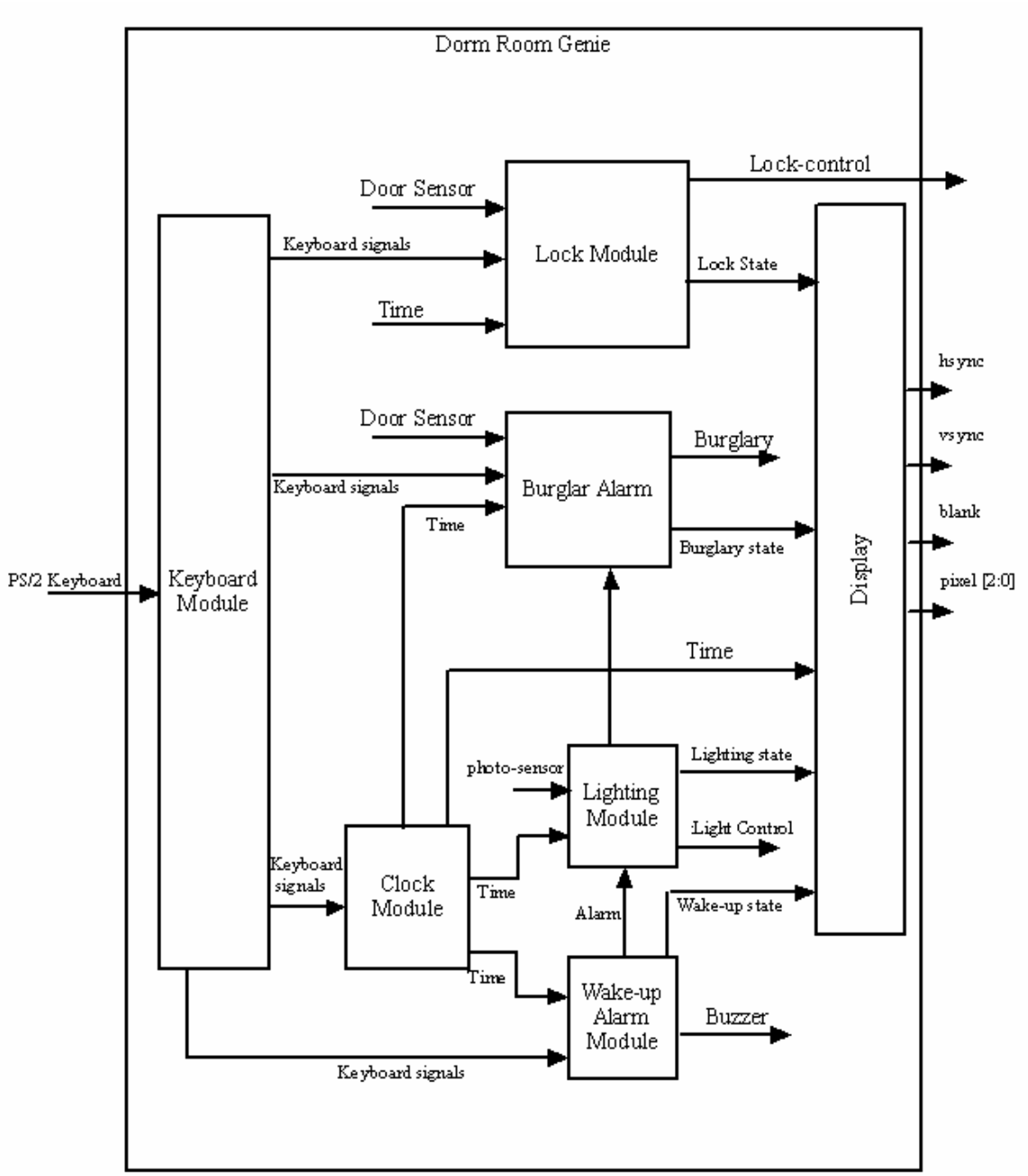
Outputs: 8-bit alpha-numeric keystroke, lock, unlock, inactive, change password, arm, disarm, set time, manual lights, automatic lights, sleep-mode, lights on, lights off, dim, brighten, set wake-up alarm, alarm acknowledge, snooze

The Display
--To be completed by David Nedzel and Aaron Stonely

The display for the user interface will show the current status of the door lock, burglar alarm, lighting control, and alarm clock features. This will allow a user to verify if their settings are correct, and make changes as needed.

Inputs: time, lock-status-display, burglar-alarm-state, wake-up state, wake-up time, lighting state, lighting brightness

Outputs: hsync, vsync, blank, [2:0] pixel (to drive the video output)


Dorm Room Genie
Door Sensor
Keyboard signals
Time
Lock Module
Lock-control
Lock State
Door Sensor
Keyboard signals
Time
Burglar Alarm
Burglary
Burglary state
PS/2 Keyboard
Keyboard Module
Display
hsync
vsync
blank
pixel [2:0]
Time
photo-sensor
Lighting Module
Lighting state
Light Control
Keyboard signals
Clock Module
Time
Alarm
Wake-up state
Time
Wake-up Alarm Module
Buzzer
Keyboard signals